

Forum:[Hack Forums](#)

Thread: [What is real safety of encrypted partition \(in file\) using VeraCrypt](#)

SuperNeo - May 23, 2021, 10:08 am

Hi. I use VeraCrypt to keep files in encrypted partition which is stored as a file (.hc) on NTFS partition. My question is: how safe it really is? If someone determined steals my computer and really wants to dig the data out... will they be able to do that and much effort will they must put into that? I don't know exactly what are mechanisms of VeraCrypt, but doesn't it just encrypt header of a partition and... that's it!?

Stetsed - May 24, 2021, 1:27 am

Well if you completed the generate random bar then first of all, brute forcing will be impossible, atleast in the age of the universe. Next to this the only real way is to bruteforce the password. So... uhhh... use a good password for it xD

Vyreazz - May 27, 2021, 1:05 pm

Better password in rarely used unicode characters

Lilith - May 28, 2021, 7:37 am

(May 27, 2021 01:05 PM)Vyreazz Wrote: [\[CITE\]](#) Better password in rarely used unicode characters Well that's just entirely misleading. ""8rLÄ_;" has an entropy of 66.60 "syjfxykMTt2h" has an entropy of 71.45. Length has just as much significance as the specific characters being used.

PerkyHash - May 28, 2021, 1:55 pm

This is an interesting one. Following

SIM-574702 - May 28, 2021, 2:30 pm

(May 28, 2021 07:37 AM)p4620389213847021 Wrote: [\[CITE\]](#) Well that's just entirely misleading. ""8rLÄ_;" has an entropy of 66.60 "syjfxykMTt2h" has an entropy of 71.45. Length has just as much significance as the specific characters being used.

I agree that length is important but I don't see how your calculation can be correct. Unicode has over 140k characters so I think you underestimated character pool size. What would your calculation of entropy for password like this be. "†‡Ž%Šv(T"

That password shows 136 bits of entropy on [This link](#) when setting it up as 140k character pool and 8 character password. If I've missed something feel free to correct me.

Lilith - May 28, 2021, 2:40 pm

(May 28, 2021 02:30 PM)matemistic Wrote: [\[CITE\]](#) I agree that length is important but I don't see how your calculation can be correct. Unicode has over 140k characters so I think you underestimated character pool size. What would your calculation of entropy for password like this be. "†‡Ž%Šv(T"

That password shows 136 bits of entropy on [This link](#) when setting it up as 140k character pool and 8 character password. If I've missed something feel free to correct me.

I was using KeePassXC's password generator. The which it shows the entropy"†‡Ž%Šv(T" to be 26.01.

Additionally, using [Kaspersky's Password Checker](#), the password you provided would take about 14

days to crack. The initial password I mentioned, ""*8rLÄ_ ;{@" would take 4 years, and "syjfykMTt2h" would take 4 centuries. [BitWarden](#) has the passwords at 3 hours, 12 days, and 3 years respectively.

SIM-574702 - May 28, 2021, 4:06 pm

(May 28, 2021 02:40 PM)p4620389213847021 Wrote: [CITE](#) I was using KeePassXC's password generator. The which it shows the entropy "†‡Ž%Šv(T" to be 26.01.

Additionally, using [Kaspersky's Password Checker](#), the password you provided would take about 14 days to crack. The initial password I mentioned, ""*8rLÄ_ ;{@" would take 4 years, and "syjfykMTt2h" would take 4 centuries. [BitWarden](#) has the passwords at 3 hours, 12 days, and 3 years respectively.

Both of those sites actually use same library to check password and it's called Zxcvbn.js. [Proof here](#) but you can look it up yourself. As far as I've seen in it's source code it doesn't support full unicode which is why "qrhdcoig" has same entropy as "qrhdcoi1" and as "†‡Ž%Šv(T". The way this library works is that it has multiple matchers but considering these passwords aren't going to be in dictionary I'll focus on brute force part.

If you take a look at this java implementation [HERE](#) you will see that it has a cardinality (estimation of character pool size) to be only 100 while unicode supports over 140k characters. That was java and I used that implementation to get basic insight in architecture but these websites use javascript so I looked into those as well and I found that cardinality is hardcoded at 10 [HERE](#). It's used [BELOW](#) and as you can see again, it doesn't factor in that character poolsize is 140k.

If we take that Entropy formula is $E = L \log_2(R)$ where R is size of the pool of unique characters from which we build the password and

L is password length, i.e., the number of characters in the password. ([Formula copied from here](#)) we can calculate entropy of "†‡Ž%Šv(T" to be $E = 8 \log_2(140k) = 136.760538413$ [Google Link](#)

My calculation is correct under assumption that whole unicode character pool is usable and no two characters are treated as same. I have not checked Veracrypt's source code but I'm confident those assumptions are true.

cokecaine - May 28, 2021, 8:29 pm

Following. A very informative thread.

SuperNeo - June 23, 2021, 1:06 pm

Hi. Maybe I don't understand something here or maybe I didn't explain myself clearly enough... so yes I filled the "random bar" with random mouse movements a lot and I have loooong password, but my question IS this: is the whole partition encrypted somehow (I don't think so) or ONLY it's header? In other words (as far as I remember) if only header is encrypted, then someone can still analyze the data of the partition, no? It would be like looking for deleted files... they are not listed in master file table, but still present on disk, so they can be found by analyzing the data byte by byte. I am right here?

BigDap - April 7, 2023, 1:47 am

interested in information, following!